Virt&I–Comm.4.2013.10

Noname manuscript No.
(will be inserted by the editor)

On the Enhancing of GriF a Workflow-oriented Grid Framework combining High-level Services

Carlo Manuali[1], **Ernesto Garcia**[2], **Antonio Laganà**[1]

Received: date / Accepted: date

Abstract The possibility of utilizing the innovative features of GriF, a Grid Framework allowing the guided management, the monitoring and the results recollection of molecular science simulations as advanced Grid Services to organize, simplify and harmonize the work carried out by computational molecular scientists has led to an enhancement of its Workflow-oriented nature and to its application to the reactive scattering components of the Grid Empowered Molecular Simulator GEMS. A key aspect considered here is the use of GriF as a Science Gateway for applications needing a search for convergence and a combination of High Throughput and High Performance computing resources. This is found in our study to provide a solid ground for the assemblage of complex computational applications when realistic (going from first principles to the simulation of the experimental signal) simulations are undertaken and for implementing Quality Evaluation mechanisms.

Keywords Grid Frameworks · Grid Molecular Simulations · Grid Services · Grid Workflow Management Systems · Quality Evaluation · Quality of Service · QoS · Quality of User · QoU · Science Gateways · Sustainability · Web Services

1 Introduction

The main difficulty for the generic users (as most of the members of the user communities are) of the Grid is the high level of computer skills still required to run distributed massive applications on it. To this end, middleware consortia (like EMI [1]) and some specific projects of the European Grid Initiative EGI [2] (like EGI-Inspire [3]) are working to develop user friendly tools.

Among these tools is GriF [4]. GriF is a Workflow-oriented Grid Framework designed within the activities of the Virtual Organization (VO) COMPCHEM [5] to provide its users with a tool allowing them to operate on the Grid and manage related operations (as for example running jobs, checking their status and retrieving related results) with no need for mastering the low-level Grid environment and using specific Grid operating system dependent commands.

To make the applications achieve a high-level of friendliness and portability, GriF has been further structured not to require the users to bear Grid Certificates (by adopting Robot Certificates [6]), to handle binary programs and to decide which Grid resources are to be used. Further enhancements of GriF have allowed operations to be carried out by the user both relying on a natural-like language (for example, when searching for an application of interest to make use of the program name or some keywords matching the desired application description and functions) and various Framework-side interventions like those related to the Grid resources match-making for the specific programs offered under the form of Grid Service. All this has been targeted to the optimization of the selection of the machines of the Grid platform with respect to their availability and to their suitability for scientific calculations by taking as a benchmark use case a specific molecular science application. For the same particular use case, the more challenging target of enabling the assemblage of highly complex Workflows (including Workflows of Workflows) has been considered.

Moreover, the enhancement of GriF has been further

[1]Department of Chemistry, University of Perugia, Perugia, Italy (IT)
[2]Department of Physical Chemistry, University of the Basque Country, Vitoria, Spain (SP)

ISSN: 2279–8773

targeted to the development of a new Quality plug-in to the end of rewarding the contributions of the COMPCHEM VO members in terms of services and activities provided either to the community as such or to individuals. This has required a preliminary study on the profiling of the VO users paving the way to a systematic evaluation of the work carried out within a Grid Community targeted to the enhancement of its Sustainability via the design of a Quality Evaluation system that is also reported here.

Accordingly, the paper is articulated as follows: in section 2 the use case (a molecular science simulator) utilized as a bench for the extensions of GriF to a high level management of the distribution of the computational applications is described in detail; in section 3 both the basic and the enhanced options offered by the GriF flowchart based on two Java servers and a Java client are illustrated; in section 4 the possibility of using the Workflow-enabled version of GriF to submit computational tasks from the Grid to external supercomputers when dealing with the so called Grid Empowered Molecular Simulator (GEMS) [7,8] from ab initio calculations to experimental observables is analyzed together with the possibility of implementing concurrent Workflows for GEMS; in section 5 the GriF plug-in allowing Quality Evaluation of Services and Users in Grid and serving as a basis to reward VO members for the work carried out on behalf of the community, is discussed. Some conclusions and ideas for future work are outlined in section 6.

2 The Use Case utilized

The structuring of GriF was prompted, as already mentioned, by the need of the users of the already mentioned COMPCHEM VO to manage on the Grid the criticalities arising when distributing GEMS [9] by avoiding being involved in the related technical aspects. The foundations of GEMS were first given in ref. [10] and implemented in a portal developed for the a priori simulation of crossed beam experiments [11] named SIMBEX. The present flow-chart of GEMS is definitely more structured than that adopted by SIMBEX. It consists, as shown by its schematic representation given in Fig. 1, of the conditional execution of the following blocks: INTERACTION, FITTING, DYNAMICS, OBSERVABLES which target, respectively, the following tasks:

- Evaluate, if not otherwise available, the electronic structure of the molecular aggregate of interest;
- Fit, using an appropriate functional form, the calculated ab initio values;

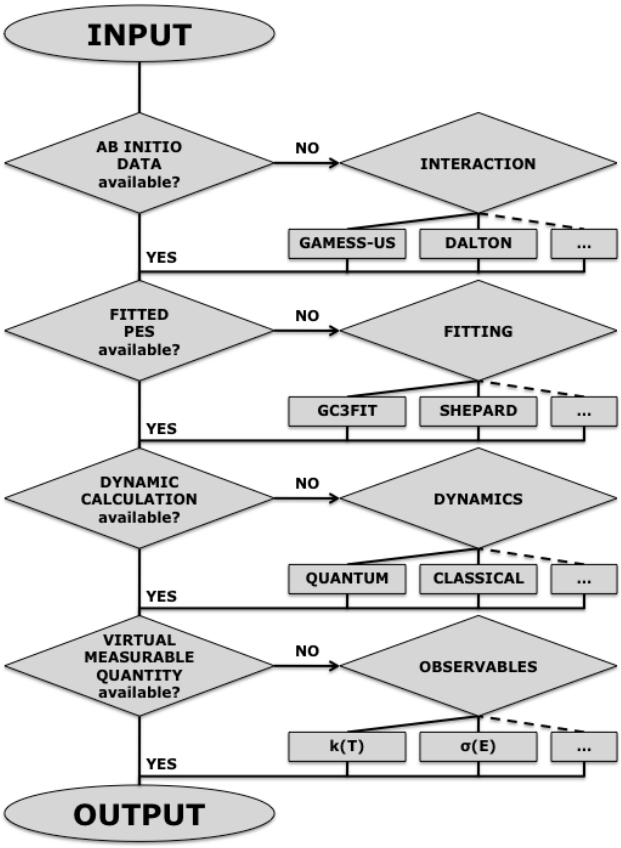

Fig. 1 The simplified flow-chart of GEMS.

- Integrate the set of scattering equations most suited for describing the motion of the nuclei of the considered system;
- Average over multiple valued parameters the scattering quantities needed to work out a theoretical estimate of the requested observable property.

The distribution of the associated tasks starts by assigning single molecular geometry ab initio calculations to each computing unit utilized by INTERACTION. The ab initio computational tasks associated with the ab initio evaluation of the electronic energy eigenvalues, $E_I(\mathbf{R})$, and eigenvectors, $\mathbf{c}_j^I(\mathbf{r};\mathbf{R})$, for the electronic Hamiltonian $\hat{H}_e$ at fixed nuclei position vectors $\mathbf{R}$:

$$\hat{H}_e \Phi_I^e(\mathbf{r};\mathbf{R}) = E_I(\mathbf{R}) \Phi_I^e(\mathbf{r};\mathbf{R}) \quad (1)$$

(with $\Phi_I^e(\mathbf{r};\mathbf{R})$ being the electronic eigenfunction expanded in a suitable basis set $\{\varphi^I(\mathbf{r})\}$ whose expansion coefficients are the already mentioned eigenvectors $\mathbf{c}_j^I(\mathbf{r};\mathbf{R})$ [12] parametrically dependent on $\mathbf{R}$) are seldom executable on the ordinary nodes of the Grid (they usually require at least clustered cpus or, even better, supercomputer facilities so as to count on large virtual memories suited to support large matrix and extended diagonalization problems). Packages already considered

for our implementations are GAMESS-US [13] and DALTON [14].
No distribution is considered for the FITTING block associated with a transformation of the pointwise representation of the interaction produced by the INTERACTION block via a procedure (usually of the least square type) that adopts a functional representation of the electronic energies easy to modify and simple to calculate. For this in-house codes tailored according to the recipes (and using when available the authors' routines) of ref. [15] and ref. [16] have been adopted in GEMS.
The most elaborated distribution schemes have been adopted for the packages of the DYNAMICS block which tackle the problem of integrating the electronically adiabatic Schrödinger equation for the nuclei wavefunction $\Psi_I(\mathbf{R},t)$ either in its Time Dependent (TD) form:

$$i\hbar\frac{d}{dt}\Psi_I(\mathbf{R},t)\Psi_I(\mathbf{R},t) = [\hat{T}_N(\mathbf{R}) + E_I(\mathbf{R})]\Psi_I(\mathbf{R},t) \quad (2)$$

(in which $\hat{T}_N(\mathbf{R})$ is the nuclear kinetic operator) or in the corresponding Time Independent (TI) version [17].
The methods implemented in GEMS are either developed in-house or available from the Web. As a matter of fact, most of the work for extending GriF was focused on the quantum mechanical atom-diatom reactive scattering codes [4,9] (for small systems) and to the classical mechanics one for the large ones [18–20]).
Typical classical mechanics packages integrate a set of first (sometimes also second) order partial differential equations starting from different sets of initial conditions. For moderately large systems these calculations are typical parameter studies applications in which a large quantity of independent tasks, each iterating over time the single time-step integration of the related classical mechanics coupled differential equations, can be easily dealt by the ordinary nodes of the Grid because requiring a limited amount of memory and an affordable (for the Grid) demand of computing time. The main problem is only the large number of events (trajectories) to be considered and the associated handling of the results storage.
Typical quantum mechanical packages integrate either in time (TD methods) [21] or in space (TI methods) [22] a large set of coupled second order partial differential equations. However, even for the simplest reactive systems, the atom diatom ones, related calculations only seldom can be re-conducted to simple parameter studies applications in which a large quantity of independent tasks, each iterating over the time (or space) step integration of the coupled differential equations, are executed. The involved differential equations, in fact, are so demanding in terms of memory (because of the size of the used matrices) that, when increasing the complexity of the system or even only seeking convergency with the total angular momentum quantum number J, the computational tasks can not be easily handled by the ordinary machines available on the Grid.
No distribution is considered for the OBSERVABLES block both because this is the most user-dependent part of GEMS and because, most often, part of its tasks is taken care already in the DYNAMICS block leaving with OBSERVABLES only the task of making an appropriate average.
To investigate the problem of an appropriate distribution of the computational tasks we have in the past performed several studies and reported related results in the literature (either by using in-house developed scripts or by exploiting the features of the P-GRADE [23] Grid Portal (see for example ref. [24])). A comparison between the use of P-GRADE and GriF shows that the latter improves on the former in efficiency and by providing the users with better operational modalities based on friendly User Driven Services. These services meet the goals of the COMPCHEM users because they allow them to easily integrate, generate and control the activities of the various computational tasks by managing the flow of the application and determining what and how data is captured and used. In particular, with the extension discussed here the users have not only the "static" possibility of deciding once for ever the packages to be composed for running (the ab initio program to use for calculating the electronic structure information, if ab initio data have not yet been produced or a potential energy routine is not already available, what type of fitting procedure has to be chosen for the ab initio data, what type of molecular dynamics code has to be adopted and, finally, what kind of experimental observable has to be simulated) yet they have the possibility of building run-time options like the insertion of barriers to synchronize computing paths, the implementation of iterations (including the condition controlled ones) to check self consistency and convergence of the calculated properties as well as the preferential routing of the tasks to specific platforms.

3 The GriF Framework

GriF has shown to be an extremely useful tool for the users because it facilitates an optimal usage of the memory, a reduced engagement of the cpu and a minimal consumption of wall time. On top of that, it leads to an optimized distribution of tasks over the network allowing the assemblage of applications of higher complexity and exploiting as well the collaboration between the experimentalists and the experts in various (complementary) theoretical areas.

ISSN: 2279–8773

Virt&I-Comm.4.2013.10

GriF is, in fact, able to capture, out of the data supplied by the Grid monitoring sensors, the information relevant to properly manage the computational applications of interest and articulate them in sequential, concurrent or alternative quality paths by adopting a Service Oriented Architecture (SOA) and Web Services [25].

3.1 The Basic Version

The SOA organization of GriF, as shown by the flow-chart given in Fig. 2, consists essentially of two Java servers and one Java client [26].

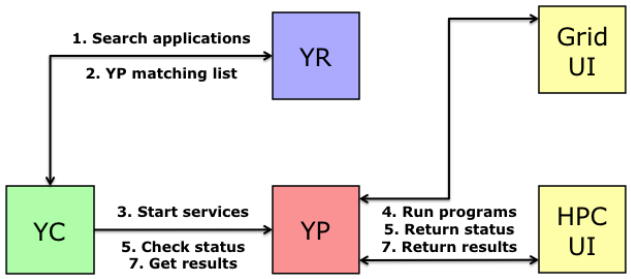

Fig. 2 The simplified flow-chart of GriF with YC being the Consumer, YR being the Registry, YP being the Provider and UIs being the entry-points to the computational platforms.

The two Java servers, called YR (acting as an information registry for available services) and YP (acting as a container providing Web Services), both describe services and reference self-describing interfaces which are published in platform-independent documents. The Java client YC takes care of running the jobs on the associated User Interface (UI) and supports the correct interfacing of the Web Services offered by GriF.
To run some programs on the Grid the user can utilize YC to perform several actions devoted to:

- The management of the basic Grid operations (e.g. checking single and multiple job status even implemented on mobile devices);
- The search of the programs on YR (e.g. searching for a YP exposing a Web Service offering the execution of the requested programs);
- The optional introduction of changes in the requested programs;
- The compilation of new executables on the selected YP by replacing the ones provided as default.

Users can also utilize the results of any previous actions to start the execution of the Grid job after passing a user-specific input file (when the default one is not appropriate) and monitor the jobs status eventually retrieving the results.

In the first implementation of GriF [27] all these features are utilized to wrap user programs, manage the Grid Certificates and Proxies, compile different executables, execute Grid jobs, monitor the jobs status and to retrieve the results.
Using GriF the members of a Grid Community are able to set up and manage YPs to expose their services in an open, standard and secure way even if having little familiarity with the wrapped programs and the Grid platform. Moreover, GriF puts the users at the center of the action as a point of generation, integration, control and evaluation of the computational activities by making use of innovative algorithms in order to provide them with more reliable computational resources and advanced running facilities representing different operational models.

3.2 Enhanced Features

As already mentioned, the original goal of GriF was to provide the COMPCHEM members with a user friendly tool allowing them to exploit the innovative features of Grid computing with no need for mastering related technicalities. In other words, GriF makes Grid applications black-box like and, at the same time, leads to a more efficient exploitation of the innovative features of the Grid when building applications of higher level of complexity. Thanks to its robust SOA framework nature GriF can, in fact, support collaboration among researchers bearing complementary expertise and contributing to collaborative work when they want to articulate their computational applications as a set of multiple tasks.
The main computational features of GriF were first exploited for running the quantum reactive scattering programs of GEMS on the Grid (such runs have allowed to carry out, for example, a systematic accurate study of the reactive efficiency of some popular atom diatom systems [28–30]). However, the problem of these codes is the fact they present a high level of complexity that needs to be properly dealt by adopting appropriate interfaces in order to efficiently support both users having a low level of expertise (training purpose) and users wishing to customize their applications for advanced studies (research purpose). To the end of distributing on the Grid massive concurrent executions, the wrapped services of GriF enabling quantum applications were improved to make use of the Collection option [31] (typical of gLite (Lightweight Middleware for Grid Computing) [32], as shown in Fig. 3).
As a matter of fact, after choosing between a custom binary program or an application already made available under the form of Grid Service, the user is requested

ISSN: 2279-8773

Virt&I-Comm.4.2013.10

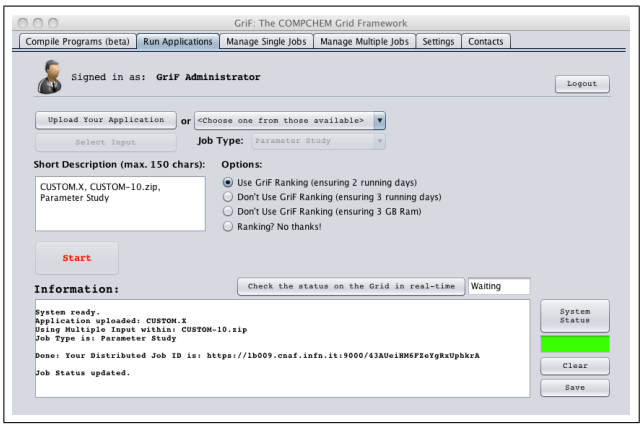

Fig. 3 The Parameter Study running modality.

to specify a compressed file in which all the different inputs (one for each subjob to be run concurrently) are stored. Accordingly, the Job Type chosen by the user will be "Parameter Study". After the job has started (by pressing the button "Start"), useful information, like the URL where to check the jobs status, are returned and each related subjob is run on the Grid, as shown in Fig. 4.

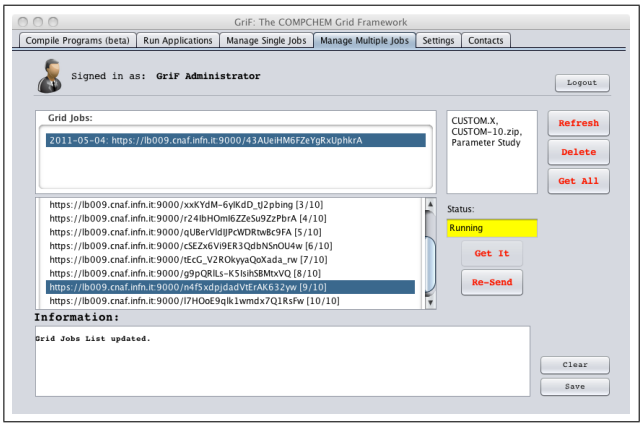

Fig. 4 The Monitoring facilities.

In this respect, for each subjob belonging to the same collection a different Computing Element (CE) queue q will be used according to the so called 'Ranking' feature of GriF. In particular, with the term 'Ranking' we define the ability of GriF to evaluate, by making use of adaptive algorithms developed by us, the Quality of each q (considering several different variables, like for example *Performance* and *Latency*) accessible by the considered Grid Community. When a user chooses to utilize the GriF Ranking feature, YP automatically selects (querying the GriF database) the q that has to be used for each subjob.

Therefore, the global Ranking of a q is given by the following general formula:

$$R(q) = R_1 + R_2 \qquad ; \qquad R_1 > 0 \tag{3}$$

where R_1 is a positive integer number representing the initial positional order of the related q resulting from the Grid itself. This is done by making use of the already available Grid middleware facilities and, in particular, by running a fake Grid job adopting some GLUE Schema [33] attributes (in particular the Requirements and Rank ones) to carry out different attempts (creating at runtime a proper Job Description Language (JDL) file specifying each time different requirements and related order, from highest to lowest values) in order to find, respectively:

1. The qs (ordered from lowest to highest cpu time) having the (maximum − current) number of running jobs > 0, the number of free cpus > 0, the number of waiting jobs equal to 0 and ensuring a wall time of 2 days at least;
2. Or (if no q matches this criterion) the qs (ordered as in the previous case) having the number of free cpus > 0, the number of waiting jobs equal to 0 and ensuring a wall time of 2 days at least;
3. Or (if no q matches this criterion) the qs (ordered as in the previous case) having the number of waiting jobs equal to 0 and ensuring a wall time of 2 days at least;
4. Or (if no q matches this criterion) the qs (ordered from the lowest to the highest number of waiting jobs) having the number of free cpus > 0 and ensuring a wall time of 2 days at least;
5. Or (if no q matches this criterion) the qs (ordered as in the previous case) ensuring a wall time of 2 days at least.

This means that, in order to be considered by GriF, a q has to be first returned by the Grid itself (in fact, in Eq. 3, $R_1 > 0$ is the necessary and sufficient condition for a q to own a $R(q)$ value and then to be considered active). In other words, by using specific Grid middleware facilities we can select which qs have to be considered. Then GriF determines the qs position in a quality-ordered rank list (as shown in Fig. 5) according to the following formula:

$$R_2 = K_P * \frac{n_f}{n_t} - \left(B[Q] + \frac{1}{K_{AL} * \frac{\sum_{i=1}^{n_d} (wt_i - ct_i)}{n_d}} \right) \tag{4}$$

where K_P is a constant weighing the ratio between the failed (n_f) and the total (n_t) number of jobs run by a q (that we identify here as the *Performance* (P) of a q) and, in the same equation, K_{AL} is a constant weighing the *Average Latency* (AL). K_{AL} is defined as the

ISSN: 2279-8773

sum over the n_d "Done" jobs of the differences between wall time (wt_i) and cpu time (ct_i) of each i "Done" job divided by n_d. B is a boxed constant depending on AL that can assume the values of a vector Q(q1, q2, q3, q4) depending on the boundary values given in a vector L(l1, l2, l3) both (arbitrarily though reasonability) chosen by the System Manager according to the resources management policy[1].

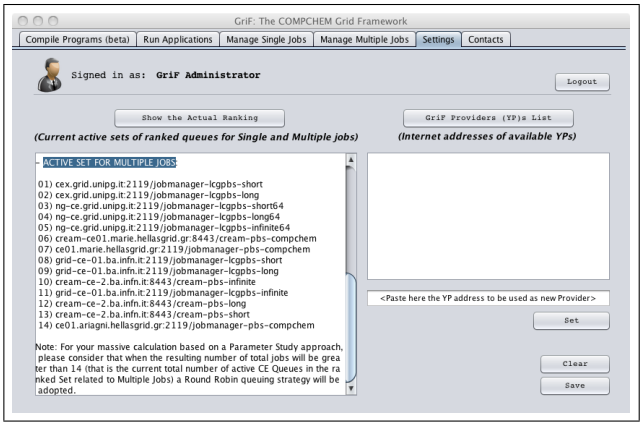

Fig. 5 The real-time Ranking of GriF.

After receiving the typical sequence of Grid messages associated with a successful completion of a job (*"Ready"*, *"Scheduled"*, *"Running"* and *"Done (Success)"*), users can access both the single (for a subjob by pressing the button "Get It") or the total (for all the available results gathered together by pressing the button "Get All") results. Moreover, users can automatically re-schedule the subjobs failed or remained in the same state for too long. This can be done by clicking the button "Re-Send" that triggers a dedicated set of Web Services responsible for the acquisition of each previous data via the Simple Object Access Protocol (SOAP) [34] protocol and their re-running by using a fresh ranked q.

4 The Workflow-enabled application of GriF

As already mentioned, the most innovative feature of the extended GriF version is its ability to support complex Workflows on Grid platforms. To illustrate its implementation we consider here as reference use case the calculation of a virtual (ab initio) observable like the detailed rate coefficient (see later for its formulation) of a molecule $M_2(v)$ in a given vibrational state v reacting with another molecule of the same type also in a given vibrational state w. Such a process is usually represented as:

$$M_2(v) + M_2(w) \rightarrow M_2(v') + M_2(w') \quad (6)$$

where, in our case, $M_2(v)$ is the diatomic molecule $N_2(v)$. The importance of such study lies in the progress in plasmas' investigations which are non equilibrium chemical systems whose individual processes experimental measurement of efficiency is difficult (if not impossible) [35]. Yet, ab initio theoretical massive computational campaigns can, in principle, provide such information. However, the cost to pay in terms of needed computer time is such to make the modeling unviable (not to mention that, in most cases, the calculations of the efficiency of electron-molecule and gas-surface elementary processes have to be added to those of processes 6). This is, indeed, the case of atmospheric reentry studies [36] in which, after all, gas phase elementary processes calculations need to be coupled with those of fluid dynamics, boundary layers and non adiabatic expansion for temperatures ranging up to 100000 K. This has made the implementation of an appropriate GriF Workflow for the study of the $N_2(v) + N_2(w)$ reaction an ideal use case.

4.1 The HPC farm section of the Workflow: the INTERACTION block

An important extension of GriF, prompted by the considered use case, is, as already mentioned, specifically tailored to take care of multi-executions of the computational tasks on the Grid following a user-defined operation logic. They provide, in fact, the user with the possibility of defining a global path to follow for the whole computational experiment to be carried out. As a matter of fact, the user has, in general, to upload first a set of instructions (say a kind of main procedure) chaining for example a given program (say A) to a following one (say B) by uploading the initial input and both A and B binary programs separately. Then he/she has to define the output of A as an input to B and, when is the case, combine the Parametric Job modality with the Workflow feature of GriF to run in parallel several subjobs using different input files and following the same operational logic. In the same way, results generated by different subjobs can be channeled via the mentioned main procedure and eventually gathered in the final result file. This means that a general Workflow supported by GriF (and defined under the form of shell script) can make use of all the available programming

[1] We are already experimenting for the next version of GriF the function:

$$B = a + b \log_{10}(AL) \quad (5)$$

where a and b are coefficients to be properly tuned in time by using either adaptive algorithms or cut-off strategies.

ISSN: 2279-8773

language constructs and produce complex sequential, conditional and iterative paths.

In this respect, the enhanced version of GriF acts as a general purpose Science Gateway [37] providing the users with the possibility of running a large variety of applications using different kinds of distribution models [38]. This applies also to the GEMS simulation considered here and allows to deal with the criticalities associated with the different utilization of the computing resources made by the various programs of the simulator when varying some parameters of the calculations or the dimensionality of the theoretical model used. Typically, as already mentioned in section 2, the distribution of GEMS on the Grid requires the use of parallel clusters or even external High Performance Computing (HPC) platforms for the INTERACTION block. As a matter of fact, most of the ab initio packages, including the already mentioned GAMESS-US and DALTON ones, when used at high level of accuracy need large sizes of memory seldom available on the Grid.

For this reason, the extended version of GriF has been equipped with a new Ranking option that offers the possibility of running a Parameter Study job by making use of a single q. In particular, the present version of GriF chooses, among those ensuring at least 1-day executions, the q having the maximum number of cpus available and therefore offering the possibility of reducing to a minimum failure probabilities. Accordingly, if the same q is able to run parallel applications exporting this information [33], it can be easily used by GriF for running local parallel programs (for example making use of MPI and OpenMP libraries) due to the fact that all the parallel communications between the subjobs will be of the intra-cluster type. At the same time, one has also the possibility of directing computational tasks to HPC machines external to the Grid. This is what has been implemented for the ab initio calculations of the INTERACTION block by activating a task farm scheme in which computational tasks were submitted to the IBM SP6 nodes of CINECA [39,40] (see Fig. 6) to run the MPI Grid-oriented version of GAMESS-US package [13] and collect back the results of the calculations.

As shown in Fig. 1 the INTERACTION block is followed by the FITTING one that is better executed on the local node due to the highly interactive nature of the related procedures (interpolations and graphical representations) needing continuous human intervention. A preliminary analysis of the ab initio results [41] shows that exchange can occur only at very high energy and that minimum energy paths leading to products' precursors prefer specific molecular geometries. However, meanwhile high level ab initio calculations are still being carried out to the end of completing a thorough investigation of the Potential Energy Surface (PES) associated with the process considered and necessary to carry out the calculations of the FITTING block, we have already undertaken DYNAMICS calculations on an available PES mainly to the end of establishing a term of comparison for the results we shall get on the new surface when ready.

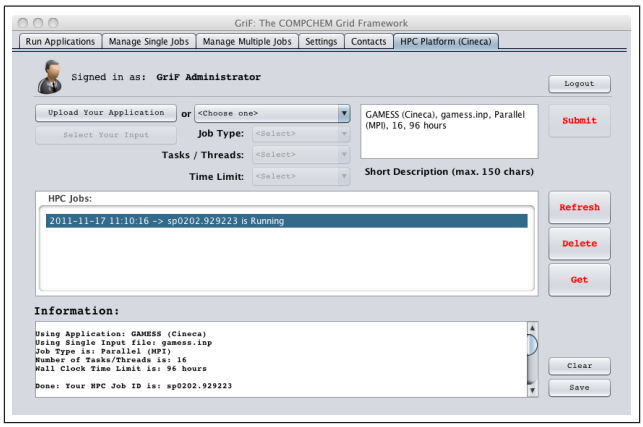

Fig. 6 HPC running.

The intramolecular component of the PES used for such calculations is made of two Morse functions taken from ref. [42] to formulate diatomic interactions. The intermolecular interaction is, instead, given by a four dimensional analytical expression based on dispersion coefficients, induction and long range electrostatic forces determined starting from dipolar moments and multipolar polarizability of both dynamics and static formulations of each monomer [43]. The quality of such PES was checked by comparing calculated and experimental second virial coefficients and integral scattering cross sections, which are found to be in good agreement. However, because of its definition the validity of such PES is confined to non-reactive processes (elastic, inelastic and partial dissociations).

4.2 The HTC farm section of the Workflow: the DYNAMICS block

Classical dynamical calculations showed us the other face of the coin when completing the structuring of the Workflow of GEMS. For the DYNAMICS block, in fact, due to the four atom nature of the $N_2(v) + N_2(w)$ reaction, the Quasi-Classical Trajectory (QCT) method (for the evolution in time the system is treated as classical while for the determination of the initial and final states a boxing of internal energy in a quantum-like fashion is made) was the technique to be used to calculate rate coefficients. The code utilized for that purpose

is VENUS96 [44]. The basic structure of VENUS96 is articulated into a first section setting the initial conditions (nature of the atoms constituting the molecular system under consideration as well as the positions and velocities of such atoms reproducing a configuration mimicking the physical status of the reactants) for a large set of trajectories to be determined by integrating a system of recursive relationships associated with the classical mechanics equations.
For the reaction considered here (see Eq. 6) this means a set of 24 Hamilton equations to be integrated using the sixth order Adams-Moulton method primed by a fourth order Runge-Kutta-Gill algorithm [45]. The integration step is set equal to 0.1 fs and allows to keep the energy conserved up to 10^{-3} kcal mol^{-1} (checked step by step). At the end of the integration the internal and collision energy of the products are evaluated. To guarantee that initial and final situations are truly asymptotic, related distances between the product molecules are set at 14 Å. Within each integration step the time consuming task is the calculation of the PES value and derivatives. This implies that each trajectory is a small memory high time consuming computational task. Accordingly, the appropriate distributed computational scheme of VENUS96 is a High Throughput Computing (HTC) task farm in which the master generates the initial conditions and the workers carry out the integration of the trajectories.
Obviously, in order to make the calculations efficient, while keeping physical significance, specific attention was put in defining the modalities of:

- setting the total number of trajectories (typically of the order of 10^8 per given temperature and pair of initial vibrational quantum numbers so as to have a reasonably low standard deviation for the quantities of interest);
- fragmenting the running of the overall number of trajectories in appropriate subsets (typically of the order of 10^4 corresponding to an average 8 hours of cpu time so as to fit the limits of most of the system queues and maximize the number of accessed ones);
- confining the number of submissions per sending (typically of the order of 300 so as to have file dimensions smaller than 200 MB requiring upload times not longer than a few minutes);
- properly chaining the random numbers of the various subsets of the fragmented distributed runs;
- implementing a large part of the results analysis within the subjobs (typically a download of the results from the Grid on the local machine so as to avoid);
- optimizing the choice of the maximum value of the impact parameter (b_{max}).

Also the DYNAMICS block is followed (like the INTERACTION one) by a highly interactive step (the OBSERVABLES block in this case). In OBSERVABLES (although some programs used in DYNAMICS may also perform part of such work) the collected set of individual trajectory results undergo a series of statistical treatments and graphical representations needing continuous human intervention. The first quantities to work out are the state (v, w) to state (v', w') thermalized at the temperature T probabilities $P(T, v, w, v', w')$:

$$P(T, v, w, v', w') = \frac{N(T, v, w, v', w')}{N(T, v, w)} \tag{7}$$

where $N(T, v, w, v', w')/N(T, v, w)$ is the fraction of trajectories that, starting from the v and w initial vibrational states (and a thermal distribution of collision and rotational energy), end up into the v' and w' final vibrational states. Such probability, though not yet an observable, is quite an illustrative property of the effectiveness of the interaction in promoting reactivity especially when plotted against the impact parameter b (the distance at which the two system would pass by in the absence of interaction) as shown in Fig. 7.

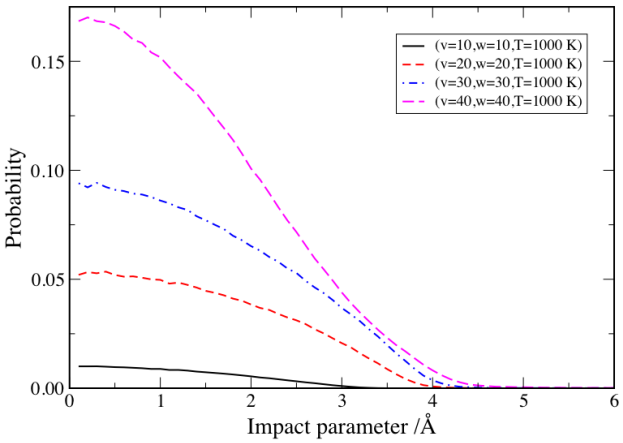

Fig. 7 Opacity function for the inelastic collisions $N_2(v) + N_2(w)$ at the temperature T.

The plot given in the figure (often called opacity function) reproduces the expected (for reactions with barrier) decreasing behavior of the state specific inelastic probabilities from a maximum at b = 0 typical of barrier dominant processes which increase with the vibrational excitation of the reactants. The state to state probability is the basic quantity for calculating observable properties like the cross section $\sigma(T, v, w, v', w')$ that is defined as:

$$\sigma(T, v, w, v', w') = \pi b_{\max}^2 P(T, v, w, v', w') \tag{8}$$

From the cross section one can also evaluate the rate coefficient $k(T, v, w, v', w')$ defined as:

$$k(T, v, w, v', w') = \sigma(T, v, w, v', w') \left(\frac{8k_B T}{\pi\mu}\right)^{1/2} \quad (9)$$

(with k_B being the Boltzmann constant and μ the reduced mass of the colliding molecules) that is the real measurable quantity of interest for the modeling.
Some illustrative values of the properly symmetrized ($k(T, v, w, v', w') = k(T, w, v, v', w')$ and $k(T, v, w, v', w') = k(T, w, v, w', v')$) rate coefficients are given in Table 1 for increasingly vibrationally excited reactants. Such preliminary results put in evidence the absence of a particular bias to favour single quantum transition as occurring in the corresponding atom-diatom reaction.

v'	*w'*	*k(T,v,w,v',w')*	*P(e)*
18	21	5.0(-14)	20%
18	22	2.0(-12)	3%
19	20	2.7(-12)	3%
19	21	7.3(-10)	0.5%
19	22	5.4(-14)	19%
20	21	2.2(-12)	3%

Table 1 Rate coefficients (in cm^3 molecule^{-1} s^{-1}) for the inelastic processes $N_2(v = 20) + N_2(w = 20) \rightarrow N_2(v') + N_2(w')$ at $T = 1000$ K (*parenthesis indicate powers of 10, P(e) indicates the percentual error*).

4.3 The Concurrency of Workflows

The above described articulation of GEMS and the diverse nature of multi-scale Molecular and Materials applications has prompted a reorganization of GriF allowing the running of complex concurrent Workflows. The one represented in Fig. 8 is the schema resulting from the implementation of the above mentioned use case that combines the two distributed sections (HPC and HTC) and the two local ones. While running it, we faced, for example, the need for launching, as already mentioned, several concurrent tasks which are themselves Workflows (say up to 500) to be managed as they were a single Grid job splitted into independent subjobs.

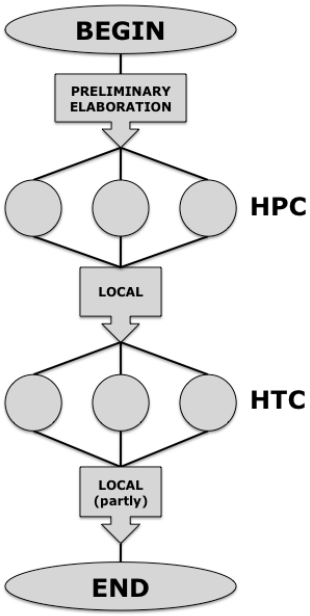

Fig. 8 The combined GriF HPC - HTC Workflow of GEMS with LOCAL being client activities, HPC being High Performance and HTC being High Throughput Grid jobs.

Accordingly, in this section a parametrization of the application allowing to interoperate via a "multiple Workflows" approach has enabled us to properly manage concurrent paths starting from different initial inputs. This led to faster composition of complex applications and to a better chaining of data results, as shown in Fig. 9.

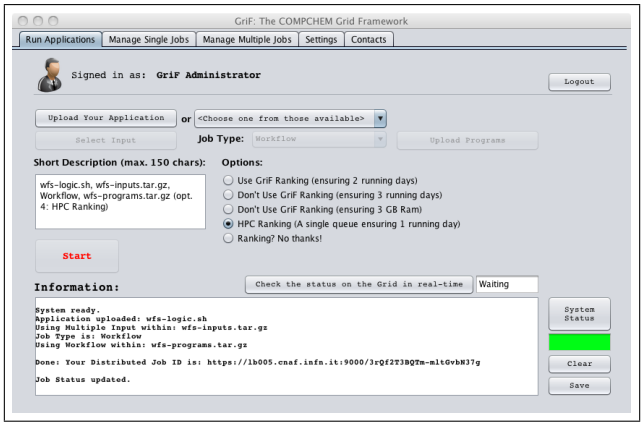

Fig. 9 Distributing Parameter Study Workflows on the Grid.

In particular, by pressing the button "Upload Your Application" the user has to firstly provide the common logic of each Workflow (under the form of Shell Script). Secondly, after choosing as Job Type the keyword 'Workflow', one can upload, respectively, a compressed package including the different single or multiple inputs (structured in sub-directories one for each independent Workflow) by using the button "Select Input". Then, one can also upload a compressed package containing all the binary programs (shared from each Workflow) involved in the computation by pressing the button "Upload Programs".
Finally, after choosing the HPC option and by pressing the button "Start" the whole distribution is submitted and the *https* identifier representing the Grid job is returned to the Information area. Finally, its extended management pursued by using properly the "Manage Multiple Jobs" panel of YC in order to delete, re-schedule and retrieve results for each Workflow (200 in the example given here, running on a different cpu of the same CE queue) is shown in Fig. 10.

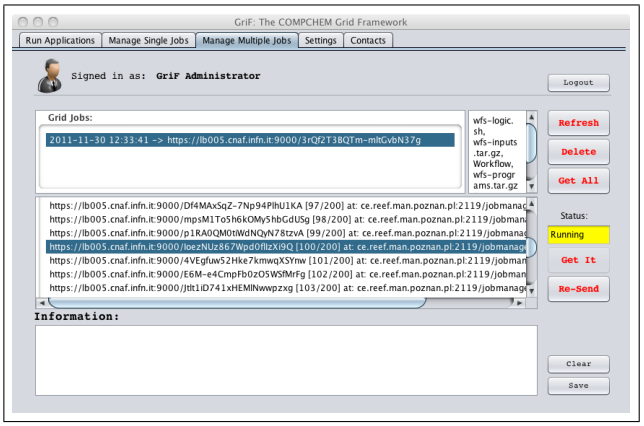

Fig. 10 Managing the Parameter Study Workflows distribution.

5 Quality Evaluations

The other innovative feature of the extended version of GriF is the Quality Evaluation (QE). QE, in fact, provides not only a means for improving the work on the Grid but also for rewarding the engagement of the COMPCHEM members in working for the VO. The adopted reward mechanisms have been designed according to a Grid Economy Model based on the evaluation of both the Quality of Service (QoS) achieved in a Grid Services-based producing modality and the Quality of User (QoU) associated with the behavior of VO members when using the Grid [46].

QoS and QoU evaluations foster, in fact, a higher level of exploitation of the Grid by:

1. enabling users to ask for Grid Services by specifying as keywords high-level capabilities rather than, for example, memory size, cpu/wall time and storage capacity;
2. allowing managers to guarantee the automatic selection of the most appropriate low-level capabilities related to the current Grid job and the adoption of different running policies (in other words, when a Grid job has to be run, GriF can make use of different system requirements according to the class level of the user owning the Grid job);
3. offering a Selection [47] (rather than a pure Discovery mechanism) based on a measure of the quality of the available Grid Services;
4. articulating the Grid Economy Model as Costs to be paid for the Services utilized and Credits earned in return for the efforts spent on behalf of the VO.

Based on this ground, GriF provides Service Orchestration features which foster the establishing of collaborative operational modalities and enhance the possibility for users and providers to cooperate in carrying out higher level of complexity computational activities leading to a gain of Credits. These Credits can then be redeemed via a preferential utilization of the resources (including the financial ones) of the VO.
It is important to point out here that in the version of GriF implemented for the present work, the concept of Service Discovery (in which VO users searching for Grid Services are provided with an unranked list of matchings) is surpassed by that of Service Selection (in which the ranking is based on QoS). At the same time, simple user quality metrics are surpassed by the Credits award mechanisms based on the new concept of QoU.
In order to quantify QoS and QoU, respectively, we have already identified, as we shall discuss later in this section, appropriate sets of non-functional properties and parameters characterizing a QoS and QoU in [46]. It is worth mentioning, however, that we have considered here as a Grid Service any set of collaborative Web Services of GriF running on the Grid by sharing a common distributed goal. As to QoU, instead, we refer to the collection and filtering of different implicit and explicit information provided by users as will be implied in the discussion below.
We want further comment that such development, in addition to leveraging on collaboration, also stimulates a significant extent of competition among the COMPCHEM members by introducing an incentive to produce innovative Grid Services and improve the existing ones. Moreover, such mechanism has also the "social" effect (within the VO) of encouraging the members of

COMPCHEM (see the articulation of membership levels adopted by COMPCHEM shown in Table 2) to further step up the quality of membership and to contribute to the infrastructure development (Hardware Provider). Credits, in fact, are also rewarded to members conferring to the VO some of their computing resources (Passive) and taking care of their deployment (Active) on the Grid (see item 3 of Table 2).

Membership Level	Short Description
1. User	*Passive*: Runs programs implemented by other VO members.
	Active: Implements at least one program for personal usage.
2. Software Provider	*Passive*: Implements at least one program for use by other members.
	Active: Manages at least one implemented program for collaborative usage.
3. Infrastructure Provider	*Passive*: Confers to the infrastructure at least a small cluster of processors.
	Active: Contributes to deploy and manage the infrastructure.
4. Manager (Stakeholder)	Takes part to the development and the management of the VO.

Table 2 Levels of Membership in COMPCHEM (*an updated version of the membership levels originally described in ref. [5] and of the Table originally published in ref. [48]* in which item 3 has been changed from "*Hardware Provider*" to "*Infrastructure Provider*").

5.1 Ranking the Quality of Services

QoS has become a significant factor in ranking the success of Grid Service providers and it plays a crucial role in Grid for estimating the usability and the utility of a Grid Service and, therefore, in determining its popularity. However, the implementation of the QoS Evaluation on the Grid is a real challenge because applications with very different characteristics and requirements compete for heterogeneous Grid resources. Moreover, with standards like SOAP, UDDI [49] and WSDL [50] being adopted by all major Web Service players, a wide range of Web Services are being currently deployed for various purposes.

Despite that, as most of the Web Services are increasingly required to comply with standards, QoS is becoming a necessary label and clearer differentiation parameters have been proposed for them. As a matter of fact, QoS relies on a whole range of criteria matching the needs of Grid users with those of the Service Providers in a competition for available Grid resources. For this reason, in our work we refer to the properties of the Web Services (such as for example Accessibility, Integrity, Performance, Reliability, Availability and Security) already used to quantify QoS [46,51].

Yet, we have also adopted, in addition, the so called *Innovation* parameter. Innovation does meet, in fact, one of the central goal of COMPCHEM and is aimed at specifically rewarding Active Software Providers (see item 2 of Table 2).

Innovation was quantified in terms of the following variables:

- *Age (A)*: number of days elapsed from the publishing date of the Grid Service considered;
- *Consolidation (C)*: a number indicating how many previous consolidated Grid Services (starting from the input of the former up to the output of the latter) have been wrapped into a new (workflow-enabled) one;
- *Diffusion (D)*: the ratio between the number of times that the Grid Service has been used and the number of VO users in a considered time interval. Diffusion implicitly comprises also the user friendliness;
- *Efficiency (E)*: the ratio between the number of results produced by the Grid Service and the number of VO users in a considered time interval;
- *Production (P)*: the ratio between a value (ranging between 0 and 1) indicating the level of new functions (e.g. with respect also to already available Grid Services recognized as standards by the scientific community of the specific domain of application considered) offered by the Grid Service (that is based on both the number and kind of them) and their related developing costs in terms of time and money also expressed by a value ranging between 0 and 1;
- *"Social" aspect (S)*: the weighed sum of three values (each of them ranging between 0 and 1) representing, respectively, the level of ethics (e.g. the promotion of universal human values like peace), of fairness (e.g. the promotion of universal availability like open source software) and of social impact (e.g. the promotion of a universal welfare) introduced by the Grid Service;
- *"Green" aspect (G)*: the amount of natural harmlessness including Energy saving (in Watt), Ecocompatibility of materials (energy saved for their materials recycling) and Space saving (in Rack Unit, or RU) of the Grid computing systems of a Hardware Provider (the Grid site).

It is worth emphasizing here that in applying QoS to COMPCHEM, we did put particular emphasis in defining the parameters S and G. For illustrative purposes we discuss in some detail here the parameter G. The steps required in order to build a "Green" Provider are

sketched in Table 3 together with its evaluation.

Steps	Objects
1a. *Design Aspects*: Collapsing	Servers -> Cpus -> Cores -> Hyper-threading technology.
	Memory modules, Hard disks, Network cards, Fans.
1b. *Design Aspects*: Virtualizing	Networks and Services.
2. *Basic HW Aspects*: Buying	Efficient hardware components (also considering RU size).
3. *Basic SW Aspects*: Enabling	Server processor power-saving features.
4. *Advanced Core Aspects*: Developing	User/System on-demand application-oriented policies to scale CPU frequencies.
5. *Human Aspects*: Saving (Energy)	Switching off monitors and unused peripherals.

Table 3 A "Green" Provider building process.

The particular attention paid to the G parameter is due to the fact that one of our sponsors (MASTER-UP srl) is active in the field of renewable energies and is developing a project in a Green chemical storage of wind energy [52].
After all, Green Computing is an emerging field of Information Technology (IT) [53] and is considered another fundamental aspect of Sustainability. Green Computing (or Green IT), in fact, refers to environmentally sustainable computing and can be defined as the study and practice of designing, manufacturing, using and disposing of computers, servers and all the associated subsystems efficiently and effectively with minimal or no impact on the environment [54]. The goals of Green Computing are similar to those of Green Chemistry: reduce the use of hazardous materials, maximize energy efficiency during the product's lifetime and promote the recyclability or biodegradability of defunct products and factory waste.
As a matter of fact, in order to determine standard evaluations for G, we refer to the SWaP (Space (S), Watt (W) and Performance (P)) formula proposed by Sun Microsystems in 2005 [55]:

$$SWaP = P\frac{S}{W} \tag{10}$$

in which one can determine P (usually measured in Giga FLOPS (FLoating point Operations Per Second)) by using standard benchmarks (as for example [56]), S by measuring the size of the Grid site in RU and W by calculating its power consumption. As an alternative, the more elaborate criterion of the GREEN500 [57] tutorial can be adopted [58].

5.2 Ranking the Quality of Users

As to QoU, the difficulty of gathering at all stages the necessary information about the users has confined their collection to a limited amount of objective data that is stored in a MySQL [59] database and combined with some user characteristics captured by the Grid middleware. This has made it possible to profile the users of the COMPCHEM VO and attempt a characterization of their QoU.
In this paper, we report as an example a tentative classification of the COMPCHEM users based on their compiling and running habits. In this respect, the QoU Evaluation parameters introduced by us [46] in order to better define the Grid users and evaluate their work [60] are essentially based on Collaborative Filtering (CF) methods [61] and in particular on the combination of the following *passive* and *active* quality information (gathered by GriF for each COMPCHEM user):

- The number of compilations (*passive*);
- The number of runs as the sum of those based on already available applications and those resulting from user compilations (*passive*);
- The number of successes over failures (*passive*);
- The number of results retrieved over available (*passive*);
- The average amount of memory consumed for runs whose results are retrieved (*passive*);
- The average cpu and wall time elapsed for runs whose results are retrieved (*passive*);
- User-specific feedbacks (*active*).

By adopting such criteria, a sample of COMPCHEM users has been first monitored in terms of their ability and/or interest to run and/or modifying the code used [62]. The results of the monitoring indicate that there are different types of users as far as the frequency of compilation with respect to submissions is concerned. This is well illustrated in Fig. 11 where the plot of the frequency of the value of the daily ratio between the number of job compilations (`Nc`) and the number of related executions (`Nx`) measured during a typical month is given.
The figure shows that the measurements are characterized by a clear bimodal structure (after dropping the sharp peak located at `Nc / Nx` $\simeq$ `0` and indicating that there is a large number of jobs runs using a one for ever compiled executable). The bimodal structure of the Figure is made of a narrow maximum located around `Nc / Nx` $\simeq$ `1` and a broader (though smaller) second one located at `Nc / Nx = 11`. The first peak refers to jobs for which the compilation is performed almost every time the job is executed whereas the second one refers to jobs for which the number of compilations largely

ISSN: 2279-8773

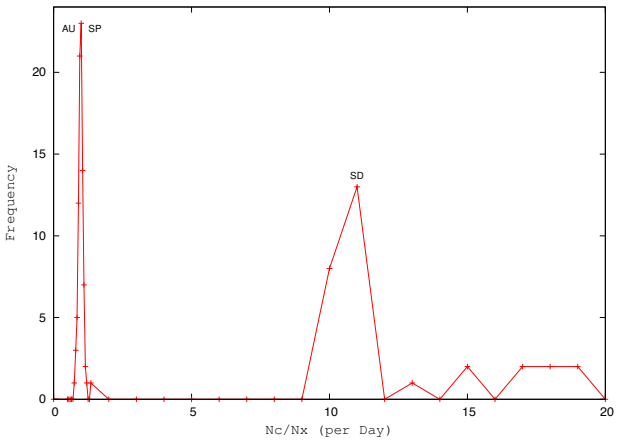

Fig. 11 The Active User (AU), Service Provider (SP) and Software Developer (SD) user profiles.

exceeds the number of runs.

The first type of jobs (associable with the dropped out near zero peak) can be easily identified as characterizing the Passive User (PU) of the first subitem of item 1 of Table 2. PUs typically run on the Grid platform a stable version of an application and, therefore, do not need to compile the application every time it is sent for execution. This type of user is more a "production scientist" (the largest and more computing intensive members of COMPCHEM) who builds his/her scientific investigations on a massive production of numerical experiments carried out using well-established numerical procedures for which extensive computational campaigns for a large variety of different conditions are needed.

The second type of jobs (associated with the lhs peak of the Figure) can be easily identified as characterizing the Active User (AU). AUs typically (see the second subitem of item 1 of Table 2) develop on the Grid platform new applications or new versions of an application. Therefore, for a certain period, they need to compile again about each time the application is sent for execution (or even more frequently) to check and validate the modifications made to the code. On the contrary, in other times, when the improved version of the application has become more stable, they will repeatedly execute it on the Grid in order to explore its range of application without compiling. This type of user is a "chemistry expert" that develops new versions of an application of the molecular science type mainly for personal use. From what it has been said, it is difficult to separate AUs from other types of COMPCHEM members whose compilation activities exceed executions. This is made apparent by the broadening of the peak around `Nc / Nx = 1` that indicates that there is an appreciable extent of variability as far as compilation before execution is concerned. This well characterizes a particular case of AU, the so called Service Providers (SP)s, which carry out software development not only for themselves but also for providing services to other users (see item 2 of Table 2). Altogether, this type of COMPCHEM members can be labeled as "chemistry expert lent to Computer Science".

The third type of jobs (associable with the rhs maximum of the Figure followed by a broad structure) can be easily identified as Software Developers (SD)s. SDs are a combination of molecular and computer scientist. This is the case, in fact, of the COMPCHEM members of the second subitem of item 3 of Table 2 having the more or less defined function of transformer computer applications into Grid Services (and testing them as well) or even acting as middleware or framework designer. For this reason, SDs bear characteristics similar (though more exasperated) to those of AU and SP. As a result, in the rhs structure of Fig. 11 the first maximum is followed by other smaller ones. This is clearly the case of a "computer scientist lent to Chemistry" for the purpose of developing COMPCHEM Grid Services by building new applications and/or new tools as opposed to the already mentioned "chemist lent to Computer Science".

Accordingly, in the first formulation of QoU (see [46, 62] our efforts have been concentrated on better defining the different user profiles of the research community members and on producing general indicators addressing for all types of users belonging to a given VO. An example of such investigation referred to the last twelve months is given in Fig. 12 where the QoU calculated for each COMPCHEM users is shown.

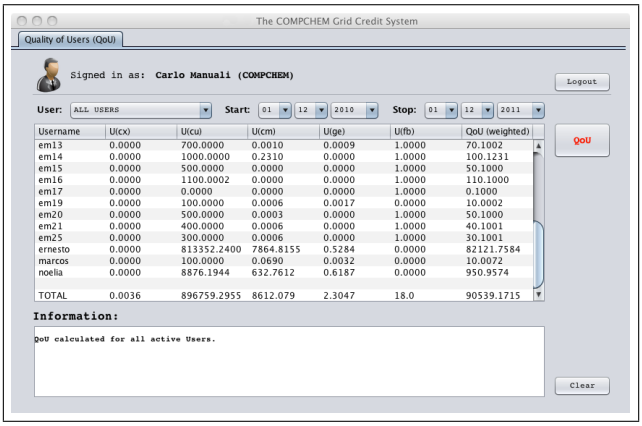

Fig. 12 Quality of User (QoU) calculated for all the COMPCHEM Users (*from December 1, 2010 to December 1, 2011*).

Such results show that QoU evaluation (and the associated Credits recognition based on the ability of GriF to properly filter qualitative and quantitative information providing a first formulation of fine QoU parameters) paves the way to the development of new structured methods and analytic models aimed at fostering Grid Sustainability.

6 Conclusions and Future Work

In the present paper an extension of GriF, a Workflow-oriented Grid Framework structuring computational applications as Web Services which can be run transparently on different computing platforms, is illustrated. The extended GriF is, now, able to compose serial and parallel tasks in a single complex Workflow and to compose as well a Workflow of Workflows. As a Use Case a prototype application of GEMS, the Grid Empowered Molecular Simulator of the COMPCHEM VO, to the study of diatom-diatom non reactive $N_2 + N_2$ processes, has been implemented.

To our knowledge this is the first time that a study of the fairly heavy nitrogen molecule nitrogen molecule reaction starting from the ab initio determination of the Potential Energy Surface and the subsequent carrying out of a trajectory dynamical study (though on a different PES) of the considered reaction has been performed using a single Grid Workflow. Results clearly show that GriF, although still in a provisory stage, is well suited for dealing with complex applications by efficiently routing Workflows of computational tasks on a single composed Workflow that routes them on different parallel and distributed machines and for offering services minimizing the required user effort while achieving maximum throughput.

The merit of the work reported here, however, goes well beyond the utility of the massive production of input data for the modeling of complex systems (like spacecraft atmospheric re-entry or cold plasmas). The introduced enhancements, in fact, highlight not only the possibility of GriF to support applications running on which of the Grid or the Supercomputer platforms are most suited for the planned massive computational campaign but also the capability of adopting a strategy combining together different Parameter Study options and Workflows.

On top of that, GriF makes it possible to rank the offered Grid services and the activities of the users on the basis of their profiles. This feature has been of particular interest because it has allowed the singling out of various types of users. Moreover, the profiling of the users has been taken as a basis to build Quality Evaluation of the work carried out in a general Grid Community (with particular reference to the COMPCHEM community) and to the award of Credits to enhance new levels of competitive collaboration.

7 Acknowledgements

The authors acknowledge interesting discussions with Marco Cecchi and Michele Carpené as well as financial support from the project EGEE III and from the COST CMST Action D37 "CHEMGRID" in particular for building the collaborative distributed network. Thanks are also due for funding to the project "Fundamental Issues on the Aerothermodynamics of Planetary Atmosphere Re-entry" (AO/1-5593/08/NL/HE) and the ESA-ESTEC contract 21790/08/NL/HE, MIUR, ARPA and MASTER-UP srl. Computer time allocation has been obtained through the COMPCHEM VO of EGI. EG acknowledges funding by the Spanish Ministry of Science and Innovation (Grant No. CTQ2008-02578).

References

1. The European Middleware Initiative (EMI); http://www.eu-emi.eu, cited 13 Dec 2011.
2. The European Grid Initiative (EGI); http://web.eu-egi.eu/documents/other/egi-blueprint, cited 13 Dec 2011.
3. The EGI-InSPIRE project (Integrated Sustainable Pan-European Infrastructure for Researchers in Europe); http://www.egi.eu/projects/egi-inspire, cited 13 Dec 2011.
4. Manuali, C., Rampino, S., Laganà, A.: GriF: A Grid Framework for a Web Service Approach to Reactive Scattering, Comp. Phys. Comm., 181, 1179-1185 (2010).
5. Laganà, A., Riganelli, A., Gervasi, O.: On the Structuring of the Computational Chemistry Virtual Organization COMPCHEM, Lecture Notes in Computer Science, 3980, 665-674 (2006); http://compchem.unipg.it, cited 13 Dec 2011.
6. INFN Certification Authority; http://security.fi.infn.it/CA/docs (2010), cited 13 Dec 2011.
7. Gervasi, O., Dittamo, C., Laganà, A.: A Grid Molecular Simulator for E-Science, Lecture Notes in Computer Science, 3470, 16-22 (2005).
8. Laganà, A., Gervasi, O.: A Priori Molecular Virtual Reality on EGEE Grid, International Journal of Quantum Chemistry, 110, 446-453 (2009).
9. Manuali, C., Laganà, A., Gervasi, O., Costantini, A.: On the Structuring of a Molecular Simulator as a Grid Service in Chemistry and Material Science Applications on Grid infrastructures, ICTP Lecture Notes Series, ISBN 92-95003-42-X, 24, 63-82 (2009).
10. Laganà, A.: Towards a grid based universal molecular simulator, in: A. Laganà, G. Lendvay (Eds.): Theory of the dynamics of elementary chemical reactions, Kluwer, Dordrecht, 363-380 (2004).
11. Gervasi, O., Laganà, A.: SIMBEX: a portal for the a priori simulation of crossed beam experiments, Future Generation Computer Systems, 20, 717-726 (2004).

12. Nebot-Gil, I., Tel, L. M., Martín, F.: Foundations of Quantum Mechanics in: Andrés, J., Bertrán, J. (Eds.): Theoretical and Computational Chemistry: Foundations, Methods and Techniques, Publicacions de la Universitat Jaume I, Castelló de la Plana, 1, 1-93 (2007).
13. The General Atomic and Molecular Electronic Structure System (GAMESS) ab initio quantum chemistry package; http://www.msg.ameslab.gov/gamess, cited 13 Dec 2011.
14. DALTON, a molecular electronic structure program, Release 2.0 (2005); http://dirac.chem.sdu.dk/daltonprogram.org, cited 13 Dec 2011.
15. Aguado, A., Tablero, C., Paniagua, M.: Global fit of ab initio potential energy surfaces: I. Triatomic systems, Computer Physics Communications, 108, 259-266 (1998).
16. Aguado, A., Paniagua, M.: A new functional form to obtain analytical potentials of triatomic molecules, Journal of Chemical Physics, 96 (2), 1265-1275 (1992).
17. Laganà, A., Riganelli, A.: Computational Reaction and Molecular Dynamics: from Simple Systems and Rigorous Methods to Large Systems and Approximate Methods in: Laganà, A., Riganelli, A. (Eds.): Lecture Notes in Chemistry, Springer Verlag, 75, 1-12 (1999).
18. VENUS, a general chemical dynamics computer program; http://monte.chem.ttu.edu/group/venus.html, cited 13 Dec 2011.
19. Gromacs; http://www.gromacs.org, cited 13 Dec 2011.
20. The DL_POLY Molecular Simulation Package; http://www.cse.scitech.ac.uk/ccg/software/DL_POLY, cited 13 Dec 2011.
21. Skouteris, D., Laganà, A., Capecchi, G., Werner, H. J.: Wave packet calculations for the Cl + H_2 reaction, International Journal of Quantum Chemistry, 99, 562 (2004).
22. Skouteris, D., Castillo, J. F., Manolopoulos, D. E.: ABC: A Quantum Reactive Scattering Program, Comp. Phys. Comm., 133, 128-135 (2000).
23. P-GRADE Grid Portal; http://portal.p-grade.hu, cited 13 Dec 2011.
24. Skouteris, D., Costantini, A., Laganà, A., Sipos, G., Balaskó, À., Kacsuk, P.: Implementation of the ABC Quantum Mechanical Reactive Scattering Program on the EGEE Grid Platform, Lecture Notes In Computer Science, 5072, 1108-1120 (2008).
25. The Web Services Architecture, W3C Working Group; http://www.w3.org/TR/ws-arch (2004), cited 13 Dec 2011.
26. Manuali, C., Laganà, A.: GriF: A New Collaborative Framework for a Web Service Approach to Grid Empowered Calculations, Future Generation Computer Systems, 27, 315-318 (2011).
27. GriF: the Grid Framework; http://www.hpc.unipg.it/grif, cited 13 Dec 2011.
28. Laganà, A., Crocchianti, S., Piermarini, V.: Towards a Full Dimensional Exact Quantum Calculation of the Li + HF Reactive Cross Section, Lecture Notes in Computer Science, 3044, 422-431 (2004).
29. Garcia, E., Carlos Sánchez, C., Saracibar, A., Laganà, A., Skouteris, D.: A detailed comparison of centrifugal sudden and J-shift estimates of the reactive properties of the N + N_2 reaction, Physical Chemistry Chemical Physics, 11, 11456-11462 (2009).
30. Rampino, S., Skouteris, D., Laganà, A.: Microscopic branching processes: The O + O_2 reaction and its relaxed potential representations, International Journal of Quantum Chemistry, 110, 358367 (2010).
31. Job Collection - EGEE SEE ROC Wiki; http://wiki.egee-see.org/index.php/Job_Collection, cited 13 Dec 2011.
32. gLite - Lightweight Middleware for Grid Computing; http://glite.cern.ch, cited 13 Dec 2011.
33. Grid: The GLUE Schema; http://www-numi.fnal.gov/offline_software/srt_public_context/GridTools/docs/glue_schema.html, cited 13 Dec 2011.
34. The Simple Object Access Protocol (SOAP) 1.2 (2007); http://www.w3.org/TR/soap, cited 13 Dec 2011.
35. Capitelli, M., Colonna, G., D'Angola, A.: Fundamental Aspects of Plasma Chemical Physics, Springer Series on Atomic, Optical and Plasma Physics, 66, Springer Science+Business Media, ISBN 978-1-4419-8182-0 (2012).
36. Armenise, I., Esposito, F., Capitelli, M.: Dissociation-recombination models in hypersonic boundary layer flows, Chem. Phys., 336, 83-90 (2007).
37. Specialised Support Centres; http://knowledge.eu-egi.eu/knowledge/index.php/Specialised_Support_Centres, cited 13 Dec 2011.
38. Foster, I., Kesselman, C.: Scaling System-Level Science: Scientific Exploration and IT implications, Computer, 39(11), 31-39 (2006).
39. SCAI (Super Computing Applications and Innovation), HPC Department of CINECA; https://hpc.cineca.it/content/about-us, cited 13 Dec 2011.
40. EGI Community Forum 2012, Munich, Germany, 26-30 March 2012; http://cf2012.egi.eu, cited 13 Dec 2011.
41. Verdicchio, M., Pacifici, L., Laganà, A.: An ab initio study of the N_2+N_2 reaction, Lecture Notes in Computer Science (*submitted*).
42. NIST Chemistry WebBook; http://webbook.nist.gov/chemistry, cited 13 Dec 2011.
43. Gómez, L., Bussery-Honvault, B., Cauchy, T., Bartolomei, M., Cappelletti, D., Pirani, F.: Chem. Phys. Lett., 445, 99-107 (2007).
44. Hase, W. H., Duchovic, R. J., Hu, X., Komornicki, A., Lim, K. F., Lu, D., Peslherbe, G. H., Swamy, K. N., Vande Linde, S. R., Varandas, A., Wang, H., Wolf, R. J.: QCPE Program, 671, Indiana University, Bloomington, Indiana.
45. Lapidus, L., Seinfeld, J. H.: Numerical Solutions of Ordinary Differential Equations, Academic Press, New York (1979).
46. Manuali, C.: Laganà, A.: A Grid Credit System Empowering Virtual Research Communities Sustainability, Lecture Notes in Computer Science, 6784, 397-411 (2011).
47. Karta, K.: An Investigation on Personalized Collaborative Filtering for Web Service, Honours Programme of the School of Computer Science and Software Engineering, University of Western Australia (2005).
48. Gervasi, O., Manuali, C., Laganà, A., Costantini, A.: On the Structuring of a Molecular Simulator as a Grid Service, in Chemistry and Material science applications on Grid Infrastructures, ICTP Lecture Notes, 24, 63-82, ISBN 929500342X (2009).
49. Universal Description, Discovery and Integration (UDDI) 3.0.2 (2005); http://www.oasis-open.org/standards, cited 13 Dec 2011.
50. Web Service Description Language (WSDL) 1.1 (2001); http://www.w3.org/TR/wsdl, cited 13 Dec 2011.
51. Mani, A., Nagarajan, A.: Understanding Quality of Service for Web Services. Improving the performance of your Web services (2002); http://www.ibm.com/developerworks/webservices/library/ws-quality.html, cited 13 Dec 2011.
52. Eos Energetics - L'accumulatore per l'energia eolica che non si ferma mai; http://www.ilsole24ore.com/art/tecnologie/2011-10-17/energetics-accumulatore-lenergia-eolica-133240.shtml, cited 13 Dec 2011.

ISSN: 2279-8773

53. The International Green Computing Conference: Sustainable Computing and Computing for Sustainability; http://www.green-conf.org, cited 13 Dec 2011.
54. Murugesan, S.: Harnessing Green IT: Principles and Practices, IEEE IT Professional, 24-33 (2008).
55. A New Metric for Server Efficiency (2005); http://www.physorg.com/news8818.html, cited 13 Dec 2011.
56. The Linpack Benchmark; http://www.top500.org/project/linpack, cited 13 Dec 2011.
57. The Green500 List: Environmentally Responsible Supercomputing; http://www.green500.org, cited 13 Dec 2011.
58. Ge, R., Feng, X., Pyla, H., Cameron, K., Feng, W.: Power Measurement Tutorial for the Green500 List (2007), http://www.green500.org/docs/pubs/tutorial.pdf, cited 13 Dec 2011.
59. The MySQL database homepage; http://www.mysql.com, cited 13 Dec 2011.
60. Wolter, N., McCracken, M. O., Snavely, A., Hochstein, L., Nakamura, T., Basili, V.: What's working in HPC: Investigating HPC User Behavior and Productivity, CTWatch Quarterly Journal, 2(4A), 9-17 (2006).
61. Su, X., Khoshgoftaar, T. M.: A Survey of Collaborative Filtering Techniques, Advances in Artificial Intelligence, 2009, article ID 421425, doi:10.1155/2009/421425 (2009).
62. Manuali, C.: A Grid Knowledge Management System aimed at Virtual Research Communities Sustainability based on Quality Evaluation, Ph.D. Thesis, Department of Mathematics and Informatics, University of Perugia (IT), 14 Feb 2011; http://www.unipg.it/carlo/PhD_Thesis.pdf, cited 13 Dec 2011.

ISSN: 2279-8773